

AMENDMENTS

In the Claims

Claims 1, and 30-54 remain unchanged from the previous amendment, which are also presented in the PENDING CLAIMS section, that begins with page 3 so as to constitute the entire set of the pending claims under consideration.

PENDING CLAIMS

1. (Unamended) A small footprint device comprising:
 - a. at least one processing element;
 - b. memory,
 - c. a context barrier for isolating one program module from at least one other program module using said memory and processing element, and
 - d. an entry point object for permitting one program module to access one other program module across said context barrier.
30. (Unamended) The small footprint device of claim 1 in which said context barrier allocates separate name spaces for each program module.
31. (Unamended) The small footprint device of claim 30 in which at least two program modules can access said entry point object even though they are located in different respective name spaces.
32. (Unamended) The small footprint device of claim 1 in which said context barrier allocates separate memory spaces for each program module.
33. (Unamended) The small footprint device of claim 32 in which at least two program modules can access said entry point object even though they are located in different respective memory spaces.

34. (Unamended) The small footprint device of claim 1 in which said context barrier enforces security checks on at least one of a principal, an object and an action.
35. (Unamended) The small footprint device of claim 34 in which at least one security check is based on partial name agreement between a principal and an object
36. (Unamended) The small footprint device of claim 35 in which at least one program can access said entry point object without said at least one security check.
37. (Unamended) The small footprint device of claim 34 in which at least one security check is based on memory space agreement between a principal and an object.
38. (Unamended) The small footprint device of claim 37 in which at least one program can access said entry point object without said at least one security check.
39. (Unamended) A method of operating a small footprint device, comprising the step of separating program modules using a context barrier and permitting access to information across the context barrier using an entry point object.
40. (Unamended) The method of claim 39, in which the context barrier will not permit a principal to perform an action on an object unless both principal and object are part of the same context unless the request is for access to an entry point object.

41. (Unamended) A method of permitting access to information on a small footprint device from a first program module to a second program module separated by a context barrier, comprising the step of creating entry point object which may be accessed by at least two program modules.
42. (Unamended) A computer program product, comprising:
- a. a memory medium; and
 - b. a computer controlling element comprising instructions for implementing a context barrier on a small footprint device and for bypassing said context barrier using an entry point object.
43. (Unamended) The computer program product of claim 42 in which said medium is a carrier wave.
44. (Unamended) A computer program product, comprising:
- a. a memory medium; and
 - b. a computer controlling element comprising instructions for separating a plurality of programs on a small footprint device by running them in respective contexts and for pennitting one program to access information from another program by way of an entry point object.

45. (Unamended) The computer program product of claim 44 in which said medium is a carrier wave.
46. (Unamended) A carrier wave carrying instructions for implementing an entry point object for bypassing a context barrier on a small footprint device over a communications link.
47. (Unamended) A carrier wave carrying instructions over a communications link for separating a plurality of programs on a small footprint device by running them in respective contexts and for permitting one program to access information from another program using at least one entry point object.
48. (Unamended) A method of transmitting code over a network, comprising the step of transmitting a block of code from a server, said block of code comprising instructions for implementing an entry point object for bypassing a context barrier on a small footprint device over a communications link.
49. (Unamended) The small footprint device of claim 1 in which said one other program module is a supercontext of said one program module.
50. (Unamended) The small footprint device of claim 1 in which the processing element runs each program module as a separate context.

51. (Unamended) The small footprint device of claim 1 in which at least one program module comprises a plurality of applets.

52. (Unamended) The small footprint device of claim 1 in which said context barrier enforces security on at least one of a principal, an object and an action.

53. (Unamended) The small footprint device of claim 52 in which at least one security check is based on partial name agreement between a principal and an object.

54. (Unamended) The method of claim 40, in which, if a principal in a first context performs an action on an entry point object in a second context, when the action is performed it will execute within the second context.

add
C1 7